Pareto Upper Confidence Bounds algorithms: an empirical study

Mădălina M Drugan
Artificial Intelligence Lab, Vrije Universiteit Brussel, Belgium, Email: madalina.drugan@vub.ac.be
Ann Nowé
Artificial Intelligence Lab, Vrije Universiteit Brussel, Belgium, Email: ann.nowe@vub.ac.be
Bernard Manderick
Artificial Intelligence Lab, Vrije Universiteit Brussel, Belgium, Email: bernard.manderick@vub.ac.be

***Abstract*—Many real-world stochastic environments are inherently multi-objective environments with conflicting objectives. The multi-objective multi-armed bandits (MOMAB) are extensions of the classical, i.e. single objective, multi-armed bandits to reward vectors and multi-objective optimisation techniques are often required to design mechanisms with an efficient exploration / exploitation trade-off. In this paper, we propose the *improved Pareto Upper Confidence Bound* (iPUCB) algorithm that straightforwardly extends the single objective improved UCB algorithm to reward vectors by deleting the suboptimal arms. The goal of the improved Pareto UCB algorithm, i.e. iPUCB, is to identify the set of best arms, or the Pareto front, in a fixed budget of arm pulls. We experimentally compare the performance of the proposed Pareto upper confidence bound algorithm with the Pareto UCB1 algorithm and the Hoeffding race on a bi-objective example coming from an industrial control applications, i.e. the engagement of wet clutches. We propose a new regret metric based on the Kullback-Leibler divergence to measure the performance of a multi-objective multi-armed bandit algorithm. We show that iPUCB outperforms the other two tested algorithms on the given multi-objective environment.**

I. Introduction

Multi-armed bandits is a machine learning paradigm used to study and analyse resource allocation in stochastic and noisy environments. The multi-armed bandit framework considers multi-objective, or multi dimensional, rewards and imports multi-objective optimisation techniques into the multi-armed bandits algorithms. We call this framework *multi-objective multi-armed bandits* (MO-MABs) [1]. Some of these techniques were also imported in other related learning paradigms [2], [3]: multi-objective Markov Decision Processes [4], [5], and multi-objective reinforcement learning [6], [7].

Multi-objective MABs lead to important differences compared to standard MABs. *Pareto dominance relation* [8] allows to maximise the reward vectors directly in the multi-objective reward space. A reward vector can optimise one objective and be sub-optimal in the other objectives, leading to many vector rewards of same quality, named incomparable. Thus, there could be several arms considered to be the best according to their reward vectors.

Like for single objective MABs, multi-objective MAB algorithms have two goals: 1) to identify the Pareto front, and 2) to minimize the regret that measures the distance between a suboptimal reward vector and the Pareto front. There are two classes of best arm identification algorithms [9]: i) with fixed budget, and ii) with fixed tolerance. A fixed budged best arm identification algorithm, i.e. successive rejects algorithm, uses a maximum of arms pulls to select the best arm. A fixed tolerance algorithm assumes that two arms can be ordered only if they are further apart then a small tolerance value. In [10], Pareto front is identified using scalarization functions and the best arm identification algorithm [11]. [12] proposed a version of Pareto front identification with fixed confidence using the Pareto ε-dominance relation [13]. In Section II, we present the multi-objective multi-armed bandits framework that uses the Pareto dominance relation.

In Section III, we introduce a Pareto UCB algorithm that deletes suboptimal arms in order to improve the upper regret bound of Pareto UCB1 [1]. This algorithm is an extension of the improved UCB algorithm [14] to reward vectors. To delete arms, their reward vectors are compared against Pareto front. This algorithm returns a set of Pareto optimal arms in a fixed budget of arm pulls. We prove an upper bound on the performance of iPUCB that is logarithmic with the number of arm pulls and the number of dimensions but also with the number of Pareto optimal arms which indicates a poor behaviour of these algorithms for a large Pareto optimal set of arms approaching the number of total arms.

We propose to use the Kullback-Leibler divergence as a frequencies based regret metric, see Section III. An ideal multi-objective multi-armed bandits would pull suboptimal arms only once and all Pareto optimal arms evenly the rest of the remaining budget. The difference between the frequencies in pulling arms with MOMAB and the frequencies resulted from an ideal policy is denoted as Kullback-Leibler regret metric.

In Section IV, we compare the performance of three MOMAB algorithms on a bi-objective stochastic environment generated with a control problem, i.e. the bi-objective wet clutch problem [15]. We show that the Kullback-Leibler regret metric is an informative regret metric for multi-objective environments optimised with MOMAB algorithms. iPUCB outperforms other two multi-objective multi-armed bandits algorithms, i.e. Pareto UCB1 [1] and Hoeffding race [16], because of the deletion of the suboptimal arms. We also show that the performance of the tested MOMAB algorithms greatly depends on the properties of the multi-objective environment. We have considered different variance of the generating Bernoulli distribution of the points in the bi-objective wet clutch problem to show that

Section V concludes this paper.

II. The multi-objective multi-armed bandits setting

Let's consider the definition of a K-armed bandit algorithm where only one arm is played at a time, where $K \geq 2$, and $I \leftarrow \{1, \ldots, K\}$ the set of arms. The objectives might be conflicting as well as correlated. In the multi-objective setting, the expected reward of each bandit i is multi-dimensional, $\mu_i = (\mu_i^1, \ldots, \mu_i^D)$, where D is a fixed number of dimensions, or objectives. When arm i is played at time steps t_1,t_2,..., the corresponding reward vectors $\mu_i^{t_1}$, $\mu_i^{t_2}$, ... are independently and identically distributed according to an unknown law with unknown expectation vector. The independence also holds between arms.

The reward vectors are ordered using the dominance relation in multi-objective spaces [8]. The following Pareto dominance relations between two reward vectors, μ and ν, are considered. A reward vector μ is considered better than, or *dominating*, another reward vector ν, $\nu \prec \mu$, if and only if there exists at least one dimension j for which $\nu^j < \mu^j$, and for all other dimensions o we have $\nu^o \leq \mu^o$. We say that μ is weakly-dominating ν, $\nu \preceq \mu$, if and only if for all dimensions j, we have $\nu^j \leq \mu^j$. A reward vector μ is considered *incomparable* with another reward vector ν, $\nu \| \mu$, if and only if there exists at least one dimension j for which $\nu^j < \mu^j$, and there exists another dimension o, for which $\nu^o > \mu^o$. We say that μ is *non-dominated* by ν, $\nu \not\succ \mu$, if and only if there exists at least one dimension j for which $\nu^j < \mu^j$.

Let *Pareto optimal reward set* $\mathcal{O}^*$ be the set of reward vectors that are non-dominated by any of the reward vectors. Let *Pareto optimal set of arms*, or Pareto front, I^* be the set of arms whose reward vectors belong to $\mathcal{O}^*$. The reward vectors in the Pareto optimal reward set $\mathcal{O}^*$ are considered equally important.

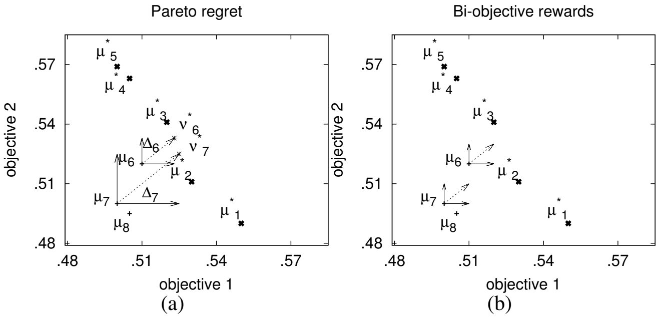
Note that Pareto dominance relation [8] is the natural order for these environments allowing to order reward vectors directly in the multi-objective reward space. However, optimisation and learning algorithms [1], [5] using Pareto dominance relations often have computational problems because of the large sets of best arms that need to be explored and stored.

A. The Pareto projection regret

The *Pareto projection* metric is introduced in [1] and measures the distance between the mean vector reward and Pareto optimal reward set $\mathcal{O}^*$, here, with a discrete number of component rewards.

To approximate the distance between $\mathcal{O}^*$ and μ_i, we construct a *virtual hypercube* with the lowest vertex is the reward vector μ_i and the upper vertex the reward vector ν_i^* that is incomparable with *all* the reward vectors in $\mathcal{O}^*$. We add to each objective μ_i a positive value ε, resulting in the upper bound of the hypercube $\nu_{i,\varepsilon}$, where $\forall j$, $\nu_{i,\varepsilon}^{*j} = \mu_i^j + \varepsilon$. The virtual optimal reward for arm i, ν_i^*, has the minimum value for ε for which $\nu_{i,\varepsilon}$ is incomparable with all rewards in $\mathcal{O}^*$. The virtual hypercube for arm i is denoted with $H(i) \leftarrow (\mu_i, \nu_i^*)$, and ε is the size of the hypercube's edge.

Fig. 1. a) Pareto projection metric for two dominated arms μ_6 and μ_7, which is Δ_6 and Δ_7, respectively. b) The dynamics of Pareto UCB1. The arm μ_6 is selected more often then the arm μ_7 because it is closer to Pareto optimal set of arms.



(a)

(b)

The Pareto projection distance of a suboptimal arm i is equal to the Euclidian distance between the virtual reward vector of arm i, ν_i^*, and the mean reward vector of the same arm, μ_i

$$\Delta_i = \sqrt{(\nu_i^* - \mu_i)^2} = \sqrt{D} \cdot \varepsilon \tag{1}$$

Since by definition ε is always positive, this distance is always positive. Note that this distance is 0 for a Pareto optimal arm since the virtual reward coincides with the optimal reward vector itself.

Example 1: Consider the eight bi-dimensional reward vectors from Figure II-A and Pareto dominance relation. Let be five optimal reward vectors, $\mu_1^* = (0.55, 0.49)$, $\mu_2^* = (0.53, 0.511)$, $\mu_3^* = (0.52, 0.541)$, $\mu_4^* = (0.505, 0.563)$ and $\mu_5^* = (0.5, 0.569)$ in $\mathcal{O}^*$. Consider three suboptimal reward vectors, $\mu_6 = (0.51, 0.52)$, $\mu_7 = (0.5, 0.5)$ and $\mu_8 = (0.505, 0.495)$. Note that the suboptimal reward μ_6 is non-dominated by three optimal reward vectors in $\mathcal{O}^*$, μ_1^*, μ_4^* and μ_5^*, but it is dominated by μ_2^* and μ_3^*. μ_7 and μ_8 are dominated by the other reward vectors.

In Figure II-A a), we show that the Pareto projection distance for two dominated mean vector rewards in Example 1 that are Δ_6 for the suboptimal arm μ_6, and Δ_7 for μ_7. By definition, $\Delta_6 = \nu_6^* - \mu_6$ and $\Delta_7 = \nu_7^* - \mu_7$. Note that Δ_6 is smaller than Δ_7 since μ_6 is closer to the Pareto front then μ_7 does. □

A *policy* π is an algorithm that chooses the next arm to play based on the list of past plays and obtained reward vectors. Let $T_i(n)$ be the number of times a suboptimal arm i has been played by the policy π during the first n plays. The expected reward vectors are computed by averaging the empirical reward vectors observed over the time. The mean of an arm i is estimated to $\widehat{\mu_i}(n) = \sum_{s=1}^{T_i(n)} X_i(s)/T_i(n)$, where $X_i(s)$ is the sample s for arm i.

The *Pareto projection cumulative regret* of π after the first n plays is the expected loss due to the play of suboptimal arms

$$\mathcal{R}_p = \sum_{i \notin I^*} \Delta_i \cdot \mathbb{E}[T_i(n)] \tag{2}$$

where $\mathbb{E}[\cdot]$ is the expectation and Δ_i is the Pareto projection distance as in Equation 1.

The *Pareto projection regret* of policy π is simply the sum of Pareto projection distances of all suboptimal arms

$$r_p = \sum_{i \notin I^*} \Delta_i \tag{3}$$

B. Pareto UCB1

Pareto UCB1 [1] is a straightforward generalisation of UCB1 where reward vectors are ranked with Pareto dominance relation. By definition, the index for each arm has two terms: i) the mean reward vector, and ii) a term related to the size of a one-sided confidence interval of the average reward according to the Chernoff-Hoeffding bounds.

As the initialisation step, each arm is played once. Each iteration, for each arm, i, we add its estimated mean reward vector and its associated confidence interval, $\widehat{\boldsymbol{\mu}}_i + \sqrt{\frac{2\ln(n\sqrt[4]{D|I^*|})}{n_i}}$. The Pareto optimal set of arms for the time step t, $I^{*(t)}$, is calculated on this index. Thus, for all not Pareto optimal arms $i \notin I^{*(t)}$, there exists a Pareto optimal arm $h \in I^{*(t)}$ that dominates arm i:

$$\widehat{\boldsymbol{\mu}}_h + \sqrt{\frac{2\ln(n\sqrt[4]{D|I^*|})}{n_h}} \succ \widehat{\boldsymbol{\mu}}_i + \sqrt{\frac{2\ln(n\sqrt[4]{D|I^*|})}{n_i}}$$

We now select uniform at random a Pareto optimal arm from $I^{*(t)}$ and pull it. After selection, the mean value of the selected arm $\widehat{\boldsymbol{\mu}}_h$ and the corresponding counters are updated. A possible stopping criteria is a fixed number of iterations n.

An arm that is closer to the Pareto optimal set of arms I^* is more often selected than an arm that is further away from I^*. Note that, by design, Pareto UCB1 is fair in selecting Pareto optimal arms.

Consider the Pareto projection distance defined in Equation 1. The expected Pareto projection cumulative regret of a policy π after any number of n plays, see Equation 2 is at most

$$\sum_{i \notin I^*} \frac{8 \cdot \log(n\sqrt[4]{D|I^*|})}{\Delta_i} + (1 + \frac{\pi^2}{3}) \cdot \sum_{i \notin I^*} \Delta_i$$

For any suboptimal arm i, the estimated time arm i is pulled is upper bounded $\mathbb{E}[T_i(n)] \leq \frac{8}{\Delta_i^2}\ln(n\sqrt[4]{D|I^*|})$ plus a small constant. Like for the standard UCB1 algorithm, the leading constant is $8/\Delta_i^2$ and the expected upper bound of the Pareto projection cumulative regret for Pareto UCB1 is logarithmic in the number of plays n. Unlike single objective UCB1, this expected bound is in addition logarithmic with the number of dimensions D and the number of Pareto optimal arms $|I^*|$. In the worst-case performance, the number of Pareto optimal arms is approximately equal to the total number of arms $|I^*| \approx K$, a probable situation in many objective environments.

Algorithm 1 Improved Pareto UCB (iPUCB)

```
Set $\tilde{\Delta}_0 \leftarrow 1$, and $B_0 \leftarrow I$
for all rounds $t = 0, 1, \ldots, \lfloor \frac{1}{2}\log_2 \frac{N}{e} \rfloor$ do
    Pull each active arm in $B_t$ for $n_t \leftarrow \left\lceil \frac{2 \cdot \log(N|I^*| \cdot D\tilde{\Delta}_t^2)}{\tilde{\Delta}_t^2} \right\rceil$ times
    Compute the Pareto front for the round $t$, $I_t^*$
    Delete all arms $i$ for which
        $\widehat{\mu}_i + \sqrt{\frac{\log(N|I^*| \cdot D\tilde{\Delta}_t^2)}{2n_t}} \prec \widehat{\nu}_i^* - \sqrt{\frac{\log(N|I^*| \cdot D\tilde{\Delta}_t^2)}{2n_t}}$
    Update the set of active arms $B_{t+1}$, and $\tilde{\Delta}_{t+1} \leftarrow \tilde{\Delta}_t/2$
end for
return the Pareto front $B_{\lfloor \frac{1}{2}\log_2 \frac{N}{e} \rfloor}$
```

III. Improved Pareto UCB

The main idea of this multi-objective MAB algorithm is inspired by the MAB algorithms that successively remove the suboptimal arms [11], [14], [17]. The best arm identification MAB algorithm from Audibert et al. [11] identifies a single optimal arm, whereas in its generalisation [18] the m-best arms are identified. Auer and Ortner [14]'s algorithm improves the upper regret bounds of UCB1 [19] by removing arms that are within a confidence interval distance from Pareto front. Note the broader goal of this algorithm is to eliminate the arms that are suboptimal rather than to identify a single optimal arm. Perchet and Rigollet [17] has a similar algorithm with a similar upper bound but adapted for multi-armed bandits with noise.

In this section, we improve the Pareto projection regret bound of Pareto UCB1 [1] by deleting arms that are not on the Pareto front. The proposed algorithm is an extension of the improved UCB algorithm [14] to reward vectors where the Pareto dominance relations are used to compare the quality of arms. The pseudo-code of the *improved Pareto UCB* (iPUCB) algorithm is presented in Algorithm 1. We assume a fixed budget N known beforehand and that the support for the underlying distribution for the reward vectors μ_i is normalised such that the regret from Equation 1 has values between 0 and 1.

At initialisation, we consider that all K arms are active, $B_0 \leftarrow I$. In each round t, the set of active arms is B_t, and each active arm is played for n_t rounds in order to compute Pareto front I_t^*, which is the set of non-dominated arms for round t. A suboptimal arm i is deleted as soon as its regret Δ_i becomes larger in each objective than a confidence value at round t, $\tilde{\Delta}_t$. The explicit condition to delete an arm i is $\widehat{\mu}_i + \sqrt{\frac{\log(N|I^*| \cdot D\tilde{\Delta}_t^2)}{2n_t}} \prec \widehat{\nu}_i^* - \sqrt{\frac{\log(N|I^*| \cdot D\tilde{\Delta}_t^2)}{2n_t}}$. Thus, an arm i is deleted if $\tilde{\Delta}_t < \frac{\Delta_i}{2}$.

Theoretical analysis. In the following theorem, we prove an upper Pareto projection regret bound by following closely the proof from [14], where Pareto dominance relation $\preceq$ is used instead of total order relation $\leq$.

Theorem 1: Let the policy in Algorithm 1 be run on a

K-armed multi-objective bandit problem, $K > 1$, having arbitrary reward distributions $\mathbf{P}_1, \dots \mathbf{P}_K$ with support in $[0,1]^D$.

The total expected Pareto projection cumulative regret of this policy up to trial N is upper bounded by

$$\sum_{i\in I:\Delta_i>\lambda}\left(\Delta_i + \frac{32\cdot\ln(N|I^*|\cdot D\Delta_i^2)}{\Delta_i} + \frac{32+64\cdot|I^*|}{\Delta_i}\right)$$

$$+\sum_{j\in I:\Delta_j\leq\lambda}\frac{64\cdot|I^*|}{\lambda} + \max_{i\in I:\Delta_i\leq\lambda}\Delta_i N$$

for all $\lambda \geq \sqrt{\frac{e}{N}}$.

Proof: This proof follows directly from the proof of Theorem 3.1 [14] and the difference is given by the usage of the Pareto dominance relations to order reward vectors. Like in [14], this proof is split in three parts corresponding to the three cases of an arm's behaviour: if the arm is a suboptimal arm or not, or if a Pareto optimal arm is eliminated or not.

Let's consider a suboptimal arm i, and let t_i be the first round in which $\tilde{\Delta}_{t_i} < \frac{\Delta_i}{2}$. By definition, we have that

$$2^{t_i} = \frac{1}{\tilde{\Delta}_{t_i}} \leq \frac{4}{\Delta_i} < \frac{1}{\tilde{\Delta}_{t_i+1}} = 2^{t_i+1} \tag{4}$$

and further that

$$\sqrt{\frac{\log\left(N|I^*|\cdot D\tilde{\Delta}_{t_i}^2\right)}{2n_{t_i}}} \leq \frac{\tilde{\Delta}_{t_i}}{2} = \tilde{\Delta}_{t_i+1} < \frac{\Delta_i}{4}$$

Let's consider that I' is the set of arms for which $\Delta_i > \lambda$ for some fixed $\lambda \geq \sqrt{e/N}$, i.e. $I' = \{i \in I \mid \Delta_i \lambda\}$.

Case (a): A suboptimal arm i is not eliminated in round t_i, and exists a Pareto optimal arm in B_{t_i}. If for an arm i in the round t_i we have

$$\widehat{\mu}_i \preceq \mu_i + \sqrt{\log\left(N|I^*|\cdot D\tilde{\Delta}_t^2\right)/2n_t} \;\;\Leftrightarrow$$

$$\forall j,\;\; \widehat{\mu}_i^j < \mu_i^j + \sqrt{\log\left(N|I^*|\cdot D\tilde{\Delta}_t^2\right)/2n_t}$$

and

$$\widehat{\nu}_i^* \succeq \nu_i^* - \sqrt{log(N|I^*|\cdot D\tilde{\Delta}_t^2)/2n_t} \;\;\Leftrightarrow$$

$$\forall j,\;\; \widehat{\nu}_i^{*j} > \nu_i^{*j} - \sqrt{\log\left(N|I^*|\cdot D\tilde{\Delta}_t^2\right)/2n_t}$$

then arm i is eliminated in the current round because

$$\widehat{\mu}_i + \sqrt{\frac{\log\left(N|I^*|\cdot D\tilde{\Delta}_t^2\right)}{2n_t}} \preceq \mu_i + 2\sqrt{\frac{\log\left(N|I^*|\cdot D\tilde{\Delta}_t^2\right)}{2n_t}}$$

$$\preceq \mu_i + \Delta_i - 2\sqrt{\log\left(N|I^*|\cdot D\tilde{\Delta}_t^2\right)/2n_t} =$$

$$\nu_i^* - 2\sqrt{\frac{\log\left(N|I^*|\cdot D\tilde{\Delta}_t^2\right)}{2n_t}} \prec \widehat{\nu}_i^* - \sqrt{\frac{\log\left(N|I^*|\cdot D\tilde{\Delta}_t^2\right)}{2n_t}}$$

Using Chernoff-Hoeffding bounds for each round $t = 0, 1, 2, \dots$ and the union bound over all objectives D, we obtain

$$I\!P\left[\widehat{\mu}_i \preceq \mu_i + \sqrt{\log\left(N|I^*|\cdot D\tilde{\Delta}_t^2\right)/2n_t}\right] \leq \frac{1}{N|I^*|\cdot\tilde{\Delta}_t^2}$$

$$I\!P\left[\widehat{\nu}_i^* \succeq \nu_i^* - \sqrt{\log\left(N|I^*|\cdot D\tilde{\Delta}_t^2\right)/2n_t}\right] \leq \frac{1}{N|I^*|\cdot\tilde{\Delta}_t^2}$$

Summing up over all Pareto optimal arms in I_t^*, the probability a suboptimal arm i is not eliminated in round t_i or before is bounded by $\frac{2}{N\tilde{\Delta}_{t_i}^2}$. The expected Pareto projection cumulative regret is the same as in [14], i.e. $\sum_{i\in B_{t_i}}\frac{2\cdot\Delta_i}{\tilde{\Delta}_{t_i}^2} \leq \sum_{i\in B_{t_i}}\frac{32}{\Delta_i}$.

Case (b): Each suboptimal arm j is either eliminated in round t (or before) or there is no Pareto optimal arm in round t_i.

Case (b_1): Assume that there exists a Pareto optimal arm in round t_i. A suboptimal arm $i \in B_{t_i}$ that is eliminated in round t_i (or before) is played not more often than

$$n_{t_i} = \left\lceil 2\cdot\log(N|I^*|\cdot D\tilde{\Delta}_{t_i}^2)/\tilde{\Delta}_{t_i}^2\right\rceil \leq$$

$$\left\lceil 32\cdot\log(N|I^*|\cdot D\Delta_i^2/4)/\Delta_i^2\right\rceil$$

times. The contribution to the total expected regret is

$$\sum_{i\in B_{t_i}}\Delta_i\left\lceil 32\cdot\log(N|I^*|\cdot D\Delta_i^2/4)/\Delta_i^2\right\rceil <$$

$$\sum_{i\in B_{t_i}}\left(\Delta_i + 32\cdot\log(N|I^*|\cdot D\Delta_i^2)/\Delta_i\right)$$

Case (b_2): Let's now consider the case where Pareto optimal arm ℓ is eliminated by a suboptimal arm i in round t_ℓ^*. Arm ℓ can only be eliminated in round t_ℓ^* by an arm i that is not yet eliminated, i.e. with $t_i \geq t_\ell^*$, where the arm i is in set $I'' = \{i \in I \mid \Delta_i > 0\}$. The corresponding contribution to the expected regret has the same value like the value from [14] for the case *(b1)*.

$$\sum_{\ell\in I_t^*}\sum_{t_\ell^*=0}^{\max_{j\in I'}t_j}\sum_{i\in I':t_i\geq t_\ell^*}\frac{2}{N\tilde{\Delta}_{t_\ell^*}^2}\cdot N\max_{j\in I':t_j\geq t_\ell^*}\Delta_j <$$

$$\sum_{\ell\in I_t^*}\left(\sum_{i\in I'}\frac{64}{\Delta_i} + \sum_{j\in I'\setminus I''}\frac{64}{\lambda}\right) = |I^*|\left(\sum_{i\in I'}\frac{64}{\Delta_i} + \sum_{j\in I'\setminus I''}\frac{64}{\lambda}\right)$$

■

Discussion. Like for the standard single objective improved UCB, the logarithmic term is the main term of this bound for an appropriate λ, e.g. $\lambda = \sqrt{\frac{e}{N}}$. The main term of the expected regret of the Pareto UCB [1] is $\frac{32\cdot\ln(N|I^*|\cdot D\Delta_i^2)}{\Delta_i}$ and it is logarithmic with the number of arm pulls, the size of the Pareto front and the number of dimensions. This bound is also smaller than the homologous term of the Pareto UCB1 algorithm. In [20] it is shown that this types of exploration/exploitation trade-off is especially useful when the regrets are close to 0.

In Example 1, the first arm deleted is μ_8^* because it has the largest virtual hypercube (i.e., the difference between the mean reward vector and the Pareto front is the largest). In the sequel, the second arm deleted is μ_7^* and the last suboptimal arm deleted is μ_6^* that has the smallest distance to the Pareto front. Ideally, iPUCB does not delete Pareto optimal arms because they have a 0 mean distance to the Pareto front. However, the data is stochastic meaning that there is variance in the mean reward vectors and a Pareto arm could be eliminated because of the variance.

A. The Kullback - Leibler divergence regret metric

The regret metric proposed in this section considers the Kullback - Leibler divergence between a perfect sampling distribution of a multi-objective multi-armed bandit, i.e. all suboptimal arms are pulled only once and all Pareto optimal arms are pulled evenly for the rest of the arm pulls, and the empirical distribution resulted from the arm pulls of a (multi-objective) multi-armed bandit algorithm.

Let p_m be the perfect frequency for a suboptimal arm i after n arm pulls, thus $p_m = \frac{1}{n}$. Let p_M be the perfect frequency for a Pareto optimal arm h after n arm pulls, thus $p_M = \frac{n-K+|I^*|}{|I^*|} \cdot \frac{1}{n}$. In total, the sum of frequencies over all arms, both Pareto optimal and suboptimal arms, is 1, where $(K-|I^*|) \cdot p_m + |I^*| \cdot p_M = 1.0$. The empirical frequency of any arm is $\widehat{p}_{m,i} = \frac{T_i(n)}{n}$ for a suboptimal arm i or $\widehat{p}_{M,h} = \frac{T_h^*(n)}{n}$ for a Pareto optimal arm h. Thus, the KL distance in term of frequencies between the empirical distribution $\widehat{p}_{m,i}$ and the perfect distribution p_m for a suboptimal arm $i \in I \backslash I^*$ after n arm pulls is given by

$$\phi_i = -p_m \cdot \ln\left(\frac{p_m}{\widehat{p}_{m,i}}\right) \tag{5}$$

The KL distance in term of frequencies between the empirical distribution $\widehat{p}_{M,h}$ and the perfect distribution p_M for a Pareto optimal arm $h \in I^*$ after n arm pulls is given by

$$\phi_h^* = -p_M \cdot \ln\left(\frac{p_M}{\widehat{p}_{M,h}}\right) \tag{6}$$

Using these KL distances, we now compute the corresponding expected cumulative and immediate regret.

The *KL cumulative regret* is defined by

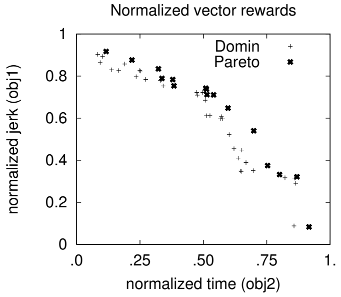
$$R_k(n) = \sum_{i \in I\backslash I^*} \phi_i \cdot \mathbb{E}[T_i(n)] + \sum_{h \in I^*} \phi_h^* \cdot \mathbb{E}[T_h^*(n)] \tag{7}$$

The *KL regret* is defined by

$$r_k(n) = \sum_{i \in I\backslash I^*} \phi_i + \sum_{h \in I^*} \phi_h^* \tag{8}$$

Note that the KL divergence metric is smaller for an algorithm that eliminates the suboptimal arms and thus gradually focuses on pulling only the Pareto optimal arms, like the improved Pareto UCB algorithm, than for an algorithm which does not eliminate the suboptimal arms, like Pareto UCB1. Thus, we consider that the Kullback - Leibler divergence regret metric can underline the performance of the multi-objective multi-armed bandits in terms of frequencies rather than only in terms of distances between mean reward vectors, like the Pareto projection regret metric does.

Fig. 2. a) The 50 bi-objective rewards representing a minimisation problem, and b) the corresponding transformation in a maximisation problem with normalised bi-objectives.



IV. EXPERIMENTS

In this section, we compare the performance of three multi-objective multi-armed bandits algorithms on a practical problem in engineering.

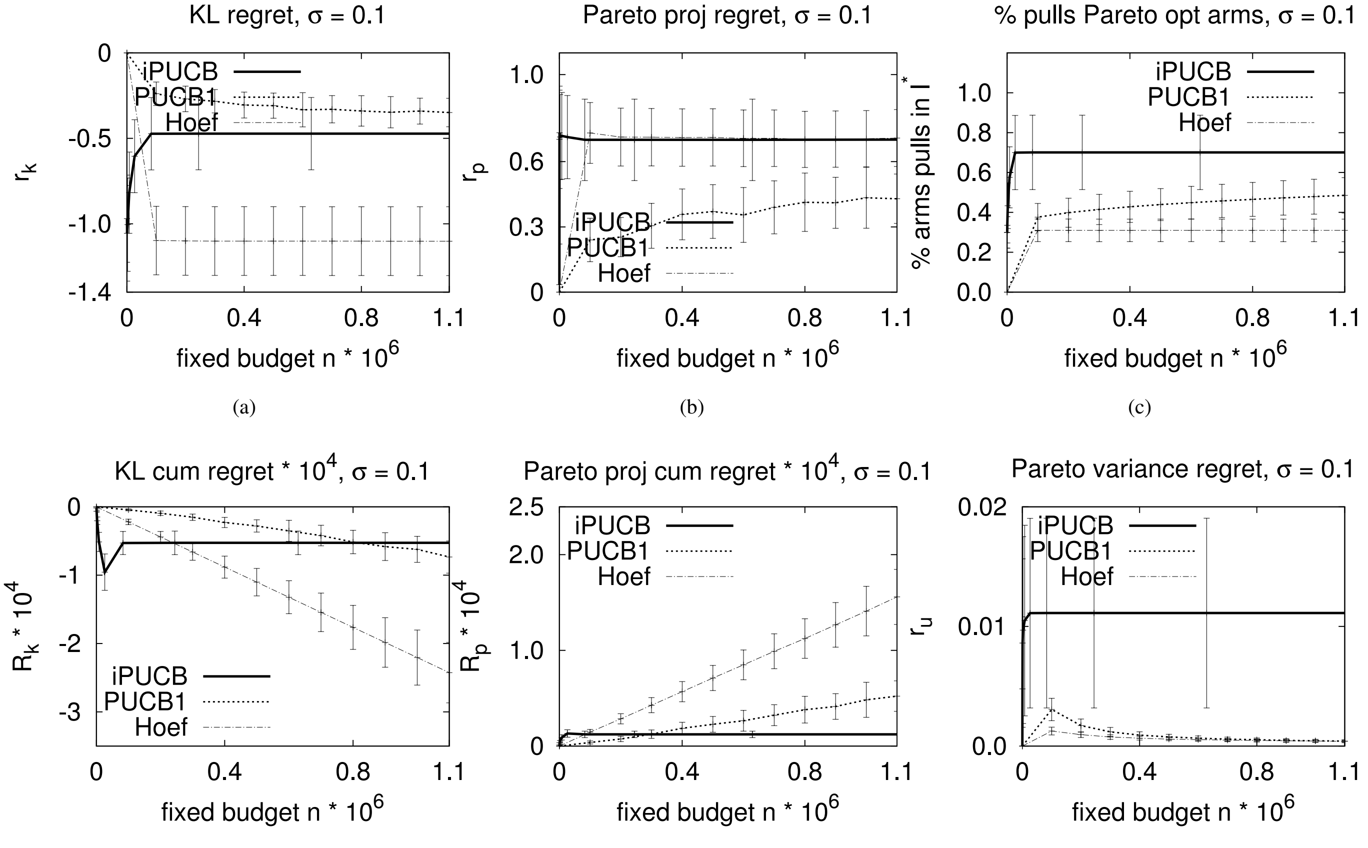
The bi-objective wet clutch problem. Our bi-objective example is a wet clutch [15] that is a system with one input characterised by a hard non-linearity when the piston of the clutch gets in contact with the friction plates. These clutches are typically used in power transmissions of off-road vehicles, which operate under strongly varying environmental conditions. The validation experiments are carried out on a dedicated test bench, where an electromotor drives a flywheel via a torque converser and two mechanical transmissions. The goal is to learn by minimising simultaneously: i) the optimal current profile to the electro-hydraulic valve, which controls the pressure of the oil to wet clutch, and ii) the engagement time. The output data is stochastic because the behaviour of the machine varies with the surrounding temperature that cannot be exactly controlled. We consider the multi-armed bandits framework adapted for multi-objective environments to find the best set of parameters for wet clutch.

In Figure 2, we show 50 points generated with this application, each point represents a trial of the machine and the jerk obtained in the given time. The original wet clutch problem is a minimisation problem that we transform into a maximisation problem, see Figure 2, by first normalising each objective with values between 0 and 1, and then transforming it into a maximisation problem for each of the two objectives.

The best set of arms with incomparable reward vectors is called the Pareto front I^* that, for this problem, is a mixture of convex and non-convex regions. There are 16 such Pareto optimal arms, which is about one-third from the total number of arms 50. Note that the suboptimal arms are very close to the Pareto front and that makes the problem difficult for multi-armed bandits especially for large variance around the mean. We have used multi-objective Bernoulli distributions with variance $\sigma = 0.1$ or $\sigma = 0.3$ for each given mean in the wet clutch problem to compare the behaviour of the given algorithms.

The tested problems. We consider three multi-objective multi-armed bandits algorithms. The baseline algorithm is

Fig. 3. Six measures are used to compare the performance of three multi-objective multi-armed bandits Hoeffding race (Hoef), Pareto UCB1 (PUCB), and improved Pareto UCB (iPUCB): a) the KL regret, b) the Pareto projection regret, c) the percentage of time one of the Pareto arms is selected, d) the KL cumulative regret, e) the Pareto projection cumulative regret, and f) the unfairness in selecting Pareto optimal arms. *The standard deviation for the multi-objective Bernoulli problem is* 0.1.



(a)

(b)

(c)

(d)

(e)

(f)

Hoeffding race (Hoef) [16] where all arms are pulled equally often. At the end, the non-dominated arms are selected.

The *Pareto UCB1* (PUCB1) algorithm [1] selects often and evenly the arms in Pareto front.

The *improved Pareto UCB* (iPUCB) algorithm proposed in Section III eliminates the arms that are not Pareto optimal when their mean is assigned with high confidence.

Settings. To compare the performance of MOMABs, we use six metrics: i) the Pareto projection regret defined in Equation 3, ii) the Pareto projection cumulative regret defined in Equation 2, iii) the KL cumulative regret defined in Equation 7, iv) the KL regret defined in Equation 8, v) the number of times one of the Pareto optimal arms is pulled, and vi) the Pareto variance regret metric defined bellow in Equation 9.

Each algorithm runs 30 times for $N = 10^6$ arms pulls. Since in practice the size of the Pareto front is unknown, we assume it to be $|I^*| \leftarrow 1$, and thus it is ignored in the calculations of the epochs length and the confidence intervals $n_t \leftarrow \left\lceil \frac{2\cdot\log(N\cdot D\tilde{\Delta}_t^2)}{\tilde{\Delta}_t^2} \right\rceil$, where $t > 0$ is an epoch. The improved Pareto UCB algorithm, cf. iPUCB, has 10 epochs of increasing length $\{40,\ 184,\ 694,\ 2478,\ 8587,\ 28927,\ 93903,\ 288269,\ 803588,\ 1.1 * 10^6\}$. From empirical observations, we notice that for a better performance of the improved Pareto UCB algorithm, the confidence interval should be multiplied with a constant 0.1, a value that is empirically set.

To define the variance in using the Pareto optimal arms, we have used the definition of unfairness as a regret metric [1]. The *Pareto variance regret* in using the Pareto optimal arms in I^* is defined as

$$r_u(n) = \frac{1}{|I^*|} \cdot \sum_{h\in I^*} \left(\widehat{p}_{M,h} - \mathbb{E}\left[\widehat{p}_{M,h}\right]\right)^2 \qquad (9)$$

where $\widehat{p}_{M,h} = \frac{T_h^*(n)}{n}$ as before. If all Pareto optimal arms are played an equal number of times, i.e. in a fair way, then $r_u(n)$ goes to 0. If a Pareto MAB-algorithm pulls only a subset of I^*, then the Pareto variance regret $r_u(n)$ is large.

A. Results

Standard deviation $\sigma = 0.1$. Figure 3 shows the performance of the three MOMAB algorithms for the bi-objective stochastic wet clutch environment with the standard variation $\sigma = 0.1$ from the bi-objective Bernoulli distribution. Figure 3 a) shows that the KL regret is closer to 0, thus better, for iPUCB1 and it is the worst for the baseline algorithm, cf.

Hoeffding race. Note that the KL regret decreases over time for iPUCB that selects often the Pareto optimal arms, but it is constant for Hoef that selects evenly all arms regardless their reward vectors. iPUCB has a constant behaviour after 10^5 arm pulls meaning that there are not any more suboptimal arms that can be deleted. In Figure 3 d), the KL cumulative regret for the improved Pareto UCB algorithm, cf iPUCB, has the smallest value after 10^6 arm pulls. It is interesting to note that the increase in the Pareto projection regret for Pareto UCB1 algorithm, cf. PUCB1, see Figure 3 b), and the constant, horizontal, slope for the other two algorithms. We explain this with the small variance for the generating bi-objective Bernoulli distribution, i.e. $\sigma = 0.1$. In contrast, the Pareto projection cumulative regret for Figure 3 e) is the smallest for the improved Pareto UCB algorithm and the largest for the Hoeffding race algorithm. To explain this behaviour, we consider the Pareto variance regret metric in Figure 3 f) and the percentage of Pareto optimal arm pulls in the total budget in Figure 3 c). Note that the variance in playing the Pareto optimal arms, cf the Pareto variance regret, is the largest for the improved Pareto UCB algorithm, cf. iPUCB, even though the same algorithm has the largest number of Pareto optimal arm pulls. This means that there is a large variance in the number of times a Pareto optimal arm is pulled in iPUCB, whereas the other two algorithms, cf. PUCB1 and Hoef, that pull less often Pareto optimal arms have consequently also less variance in pulling these arms.

Standard deviation $\sigma = 0.3$. In Figure 4, we show that performance of the tested MOMABs for an increased standard deviation of the generating bi-objective Bernoulli distribution $\sigma = 0.3$. Due to the large noise around the mean values, it is more difficult to correctly disseminate between the Pareto optimal arms and the suboptimal arms that are close to the Pareto front. The improved Pareto UCB algorithm, cf iPUCB, has the worst the KL regret metric in Figure 4 a) but the best KL cumulative regret metric in Figure 4 d). The variance in pulling Pareto optimal arms, Figure 4 f), also increases for the tested MOMAB algorithms except for the Hoeffding race that pulls evenly all arms. The Pareto projection regret metric in Figure 4 b) and e) is better, thus smaller, than the homologue in Figure 3 b) and e) for the improved Pareto UCB and Pareto UCB1 algorithms because of the mean of the two environments are unchanged and the variance of these environments increases. In Figure 4 c), the percentage of Pareto optimal arm pulls is smaller than for a smaller variance $\sigma = 0.1$, and the Pareto variance regret is larger than a larger variance around the mean.

Discussion. To conclude, the improved Pareto upper confidence bound algorithm, cf. iPUCB, is the best performing algorithm for the tested cumulative regret metrics because it eliminates the suboptimal arms. The Pareto UCB1 algorithm, cf. PUCB1, is the second best performing algorithm because it selects often the Pareto optimal arms. Both algorithms outperform the baseline algorithm, cf. the Hoeffding race algorithm.

The improved Pareto upper confidence bound algorithm is also the most robust algorithm given the stochastic variance of the bi-objective environment, i.e. its performance varies less with the change in standard deviation. Comparatively, the difference in the percentage of Pareto optimal arm pulls between Figure 3 c) and Figure 4 c) is smaller for the improved Pareto upper confidence bound, cf iPUCB, than for the Pareto UCB1, cf PUCB1. It is interesting to note that even though the performance of the Hoeffding race algorithm is invariant with the value of the standard deviation, the corresponding regret metrics change their mean values with an increase in the standard deviation for these values. Thus, as expected, there is not statistical significance change for the Hoeffding race algorithm in the two settings.

The KL regret metric, both the immediate and the cumulative variants, have a different behaviour than the Pareto projection regret metric for the tested environments. The tested environment has many suboptimal arms that are very close to the Pareto front and thus the corresponding Pareto projection distances are rather small for all arms. This is exactly the situation where the Pareto UCB1 algorithm that select arms proportionally with their mean reward vector will behave poorly and thus the improved Pareto UCB algorithm can ameliorate the performance of Pareto UCB1 algorithm.

V. Conclusions

We introduce a variant for the multi-objective multi-armed bandits algorithm with multiple, possibly conflicting, reward values, i.e. the improved Pareto upper confidence bound algorithm, that eliminates the suboptimal arms in order to identify the Pareto optimal set of arms. The improved Pareto UCB algorithm compares each suboptimal arm against the Pareto front and deletes the suboptimal arms when their mean is assigned with given confidence. We prove logarithmic upper regret bounds for the improved Pareto UCB algorithm and we propose a KL regret metric as a performance metric for the experiments. We compare the proposed multi-objective multi-armed bandits algorithm on a bi-objective Bernoulli reward distribution inspired by a real-world problem, i.e. the wet clutch. To conclude, we show both empirically and theoretically that the improved Pareto UCB algorithm is an efficient alternative to Pareto UCB1 algorithm.

References

[1] M. Drugan and A. Nowe, "Designing multi-objective multi-armed bandits: a study," in *Proc of International Joint Conference of Neural Networks (IJCNN)*, 2013.

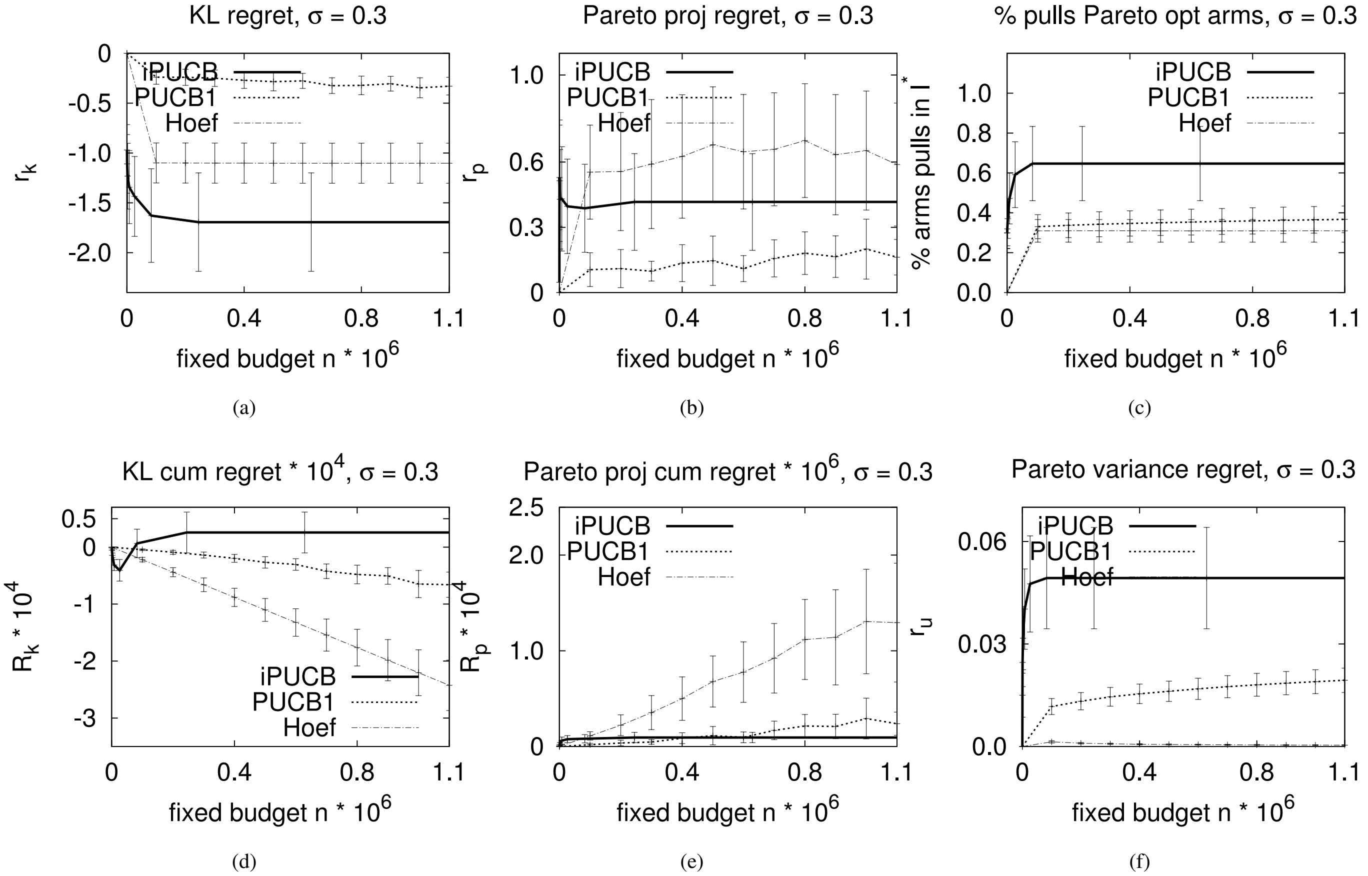
[2] P. Vamplew, R. Dazeley, A. Berry, R. Issabekov, and E. Dekker, "Empirical evaluation methods for multiobjective reinforcement learning algorithms," *Machine Learning*, vol. 84, no. 1-2, pp. 51–80, 2011.

[3] D. M. Roijers, P. Vamplew, S. Whiteson, and R. Dazeley, "A survey of multi-objective sequential decision-making," *J. Artif. Intell. Res. (JAIR)*, vol. 48, pp. 67–113, 2013.

[4] D. J. Lizotte, M. Bowling, and S. A. Murphy, "Efficient reinforcement learning with multiple reward functions for randomized clinical trial analysis," in *Proceedings of the Twenty-Seventh International Conference on Machine Learning (ICML)*, 2010.

[5] M. A. Wiering and E. D. de Jong, "Computing optimal stationary policies for multi-objective markov decision processes," in *Proc of Approximate Dynamic Programming and Reinforcement Learning (ADPRL)*. IEEE, 2007, pp. 158–165.

Fig. 4. Six measures used to compare the performance of three multi-objective multi-armed bandits Hoeffding race (Hoef), Pareto UCB1 (PUCB), and improved Pareto UCB (iPUCB) when the standard deviation for the multi-objective Bernoulli problem increases to 0.3.



(a)

(b)

(c)

(d)

(e)

(f)

[6] K. van Moffaert, M. M. Drugan, and A. Nowe, “Hypervolume-based multi-objective reinforcement learning,” in *Proc of Evolutionary Multi-objective Optimization (EMO)*. Springer, 2013.

[7] W. Wang and M. Sebag, “Multi-objective Monte Carlo tree search,” in *Asian conference on Machine Learning*, 2012, pp. 1–16.

[8] E. Zitzler, L. Thiele, M. Laumanns, C. M. Fonseca, and V. da Fonseca, “Performance assessment of multiobjective optimizers: An analysis and review,” *IEEE T. on Evol. Comput.*, vol. 7, pp. 117–132, 2003.

[9] V. Gabillon, M. Ghavamzadeh, and A. Lazaric, “Best arm identification: A unified approach to fixed budget and fixed confidence,” in *NIPS*, 2012, pp. 3221–3229.

[10] M. M. Drugan and A. Nowe, “Scalarization based pareto optimal set of arms identification algorithms,” in *Proc of International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2014.

[11] J.-Y. Audibert, S. Bubeck, and R. Munos, “Best arm identification in multi-armed bandits,” in *Proc of Conference on Learning Theory (COLT’10)*, 2010.

[12] M. M. Drugan and A. Nowe, “ε-approximate pareto optimal set of arms identification in multi-objective multi-armed bandits,” in *Proc of Belgian-Dutch conference on Machine Learning (BENELEARN)*, 2014.

[13] M. Laumanns, L. Thiele, K. Deb, and E. Zitzler, “Combining convergence and diversity in evolutionary multiobjective optimization,” *Evolutionary Computation*, vol. 10, no. 3, pp. 263–282, 2002.

[14] P. Auer and R. Ortner, “UCB revisited: Improved regret bounds for the stochastic multi-armed bandit problem,” *Periodica Mathematica Hungarica*, vol. 61, no. 1-2, pp. 55–65, 2010.

[15] K. V. Vaerenbergh, A. Rodriguez, M. Gagliolo, P. Vrancx, A. Nowe, J. Stoev, S. Goossens, G. Pinte, and W. Symens, “Improving wet clutch engagement with reinforcement learning,” in *International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2012.

[16] O. Maron and A. Moore, “Hoeffding races: Accelerating model selection search for classification and function approximation,” in *Advances in Neural Information Processing Systems*, vol. 6. Morgan Kaufmann, 1994, pp. 59–66.

[17] V. Perchet and P. Rigollet, “The multi-armed bandit problem with covariates,” *The Annals of Statistics*, vol. 41, no. 2, pp. 693–721, 2013.

[18] S. Bubeck, T. Wang, and N. Viswanathan, “Multiple identifications in multi-armed bandits,” in *Proc of International Conference on Machine Learning (ICML’13)*, 2013.

[19] P. Auer, N. Cesa-Bianchi, and P. Fischer, “Finite time analysis of the multiarmed bandit problem.” *Machine Learning*, vol. 47, no. 2/3, pp. 235–256, 2002.

[20] S. Bubeck and N. Cesa-Bianchi, “Regret analysis of stochastic and nonstochastic multi-armed bandit problems,” *Foundations and Trends in Machine Learning*, vol. 5, no. 1, pp. 1–122, 2012.